Estimating Probabilities of Default using Support Vector Machines

A Master Thesis Presented

by

Sebe-Vodislav Razvan-Alexandru

(513295)

to

Prof. Dr. Wolfgang Härdle

CASE - Center of Applied Statistics and Economics

Humboldt University, Berlin

and

Dr. Rouslan Moro

in partial fulfillment of the requirements

for the degree of

Master of Science

Berlin, 13. August, 2009

Abstract

Optimizing capital allocation by better estimating probability of default requires generally new model selection.

An analysis of German solvent and default companies was performed using the promising Support Vector Machines (SVM) methodology. The analysis shows good performance of the SVM compared to the Logit model with respect to the accuracy indicators. Also, the SVM scores enable the estimation of probabilities of default for new companies.

Contents

1 Introduction

Estimating default probabilities was, is and will always be a hot topic that will attract attention due to its importance and effects. The high number of articles on default probability in academia and the role of risk management within companies confirms the statement.

Generally, the probability of default is the likelihood that a company or natural person will not be able to pay on his loan or his debt. Also, the probability of default is one of the parameters used in the Basel II agreement for the calculation of regulatory capital for a banking institution. There are 2 common capital ratios banking institution must fulfil, namely:

$$Tier\ 1\ capital\ ratio\ = \frac{Tier\ 1\ capital}{Risk-adjusted\ assets} \geq 6\% \tag{1}$$

and

$$Total\ capital\ ratio\ = \frac{Total\ capital\ (Tier\ 1\ and\ Tier\ 2)}{Risk-adjusted\ assets} \geq 10\% \tag{2}$$

For the calculation of the risk-weighted assets a bank may apply the simple risk weight approach (SRWA) or the internal models approach (IMA) which uses own estimates of probabilities of default.

Furthermore, banks must have adequate provisions to cover the expected losses that may encounter in the lending activity as we see in Figure 1. The provision for a loan is equal to the expected loss of that loan and uses the probability of default for

its calculation as follows:

$$EL = EAD \ * \ PD \ * LGD \tag{3}$$

where, EL is the expected loss, EAD is the exposure at default, PD is the probability at default and LGD is the loss given default.

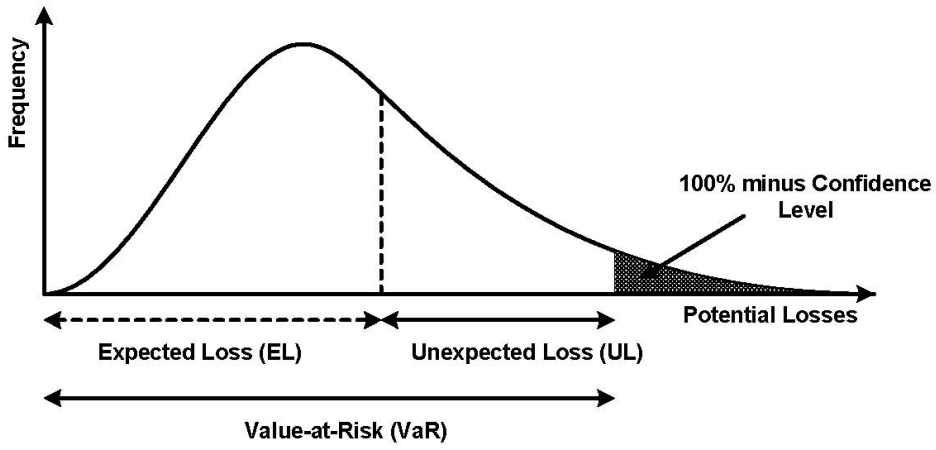


Figure 1: Loss distribution

Moreover, banks use default probabilities to determine the solvency of their business partners and rating agencies make use of default probabilities to assess the risk class of different companies.

The classical statistical techniques in the literature are discriminant analysis (DA) and the logit model. However, statistical score analysis has older roots. According to Thomas et al. (2002) in the 1930's data mining was used when companies introduced numerical score cards for their clients. Later on, in 1966, Beaver came up with the DA model for univariate case. It was followed by Altman's Z score in 1968 with an

expansion to the multivariate case:

$$Z = v_1 * x_1 + v_2 * x_2 + \ldots + v_n * x_n \tag{4}$$

where v_i are the weights and x_i are the input indicators, which originally were the following:
$x_1 = \frac{Working\ capital}{Total\ assets}$, $x_2 = \frac{Retained\ earnings}{Total\ assets}$, $x_3 = \frac{EBIT}{Total\ assets}$, $x_4 = \frac{Market\ value\ equity}{Book\ value\ of\ total\ debt}$, $x_5 = \frac{Sales}{Total\ assets}$.

The logit and probit models were introduced in Martin (1977), Ohlson (1980) and soon replaced the old-fashioned DA. The main difference between the DA and the logit model is that the logit model does not assume multivariate normality and equal covariance matrices as in the DA case. We may say, from this point of view, that the logit model is a generalisation of the DA model.

In time the demand for more accurate default estimations and the availability of bigger databases led to the research of other complex and highly quantitative estimation methods such as: artificial neural nets (ANN), decision trees, general algorithms and support vector machines (SVM).

The purpose of this paper is to compare the efficiency of the SVM with the logit model and to come up, as the title suggests, with a method for estimating the probability of default using support vector machines.

The paper is structured as follows: in the second chapter a theoretical presentation of the SVM will be introduced. The database and the analysis methodology will be presented in the third chapter, followed by the empirical results in the fourth chapter. Finally, the last chapter will be used for observations and conclusions.

2 Support Vector Machines (SVM) Methodology

The SVM technique came to the academic attention in the 1990's, in Vapnik (1995) and Vapnik (1997) when the idea of quadratic programming optimization took form and modern software was available for complicated computations.

The support vector machines approach is one of the new learning methods used in binary classification and it implements the following idea: it maps the input vectors into a high-dimensional feature space Z through some non-linear mapping, chosen a priori, where an optimal hyperplane may separate the 2 groups of subjects.

Support vector machines have been applied successfully in many classification problems such as text categorisation, image recognition and gene expression analysis according to Cristianini and Shawe-Taylor (2000).

Using SVM in credit scoring came as a natural step in statistics and finance and has the following story as starting point: we have information about n input vectors: x_i, $i = 1, \ldots, n$ that represent companies and that contain financial indicators such as *Return on Assets* (ROA), *Return on Equity* (ROE), *Leverage*, etc. Also, we have n indicator output vectors y_i, $i = 1, \ldots, n$ that give us information about whether the company is solvent or not:

$$\mathrm{y}_i = \begin{cases} 1 & if \ \mathrm{x}_i \text{ default} \\ -1 & if \ \mathrm{x}_i \text{ non default} \end{cases}$$

The x_i define a space of labelled points which is called input space.

The idea is to find a separating hyperplane that maximizes the margin of the two

data classes. The margin is defined as the minimal distance between the hyperplanes that bound each class. Later on, by using the weights that define the separating hyperplane we can obtain the decision function for new observations.

The bounds, the separating hyperplane and the error are illustrated in Figure 2.

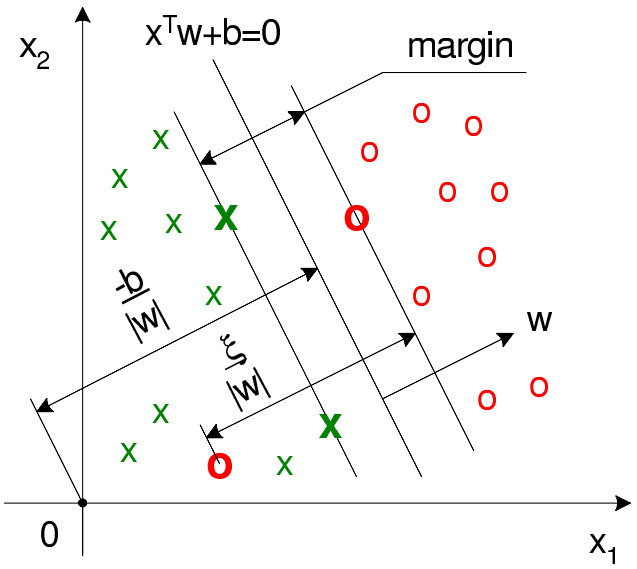


Figure 2: Separating hyperplane

The points that are on the hyperplanes are called support vectors since they are vectors in an n-dimensional input space and also because they support the position where the hyperplane lies. Whereas the other points are called non-suppport vectors since they are vectors and by removing them our separating hyperplane will not change its configuration.

As we have seen above, we deal with binary classification and the points are labelled either +1 or -1.

Going from the geometrical picture to the mathematical approach, the separating

hyperplane is represented so:

$$w^\top x + b = 0 \tag{5}$$

where b is the bias or trashhold, w is the weight vector and x is the data vector.

The separating hyperplane will have the following properties:

$$(w^\top x_i) + b > 0 \ , if \ y_i = 1 \tag{6}$$

$$(w^\top x_i) + b < 0 \ , if \ y_i = -1 \tag{7}$$

Then we will aim at finding a decision function that may interpret new data:

$$D(x) = sign(w^\top x + b) \tag{8}$$

We will start with the linear separable data case using the equations of the upper and lower bounds for the support vectors on both sides:

$$w^\top x_{upper} + b = 1 \tag{9}$$

and

$$w^\top x_{down} + b = -1 \tag{10}$$

By subtraction, we get the following:

$$w^\top (x_{upper} - x_{down}) = 2 \tag{11}$$

The margin is given by the projection of a vector $(x_{upper} - x_{down})$ onto the normal vector to the hyperplane which leads to:

$$\frac{2}{\|w\|} = \frac{2}{\sqrt{w^\top w}} \tag{12}$$

Then, we transform the quadratic maximization problem into a minimization problem by taking the inverse and since square root is a monotonic function we get rid of it:

$$\min_{w,b} \frac{1}{2} w^\top w \tag{13}$$

subject to:

$$y_i(w^\top x_i + b) \geq 1 \tag{14}$$

$$i = 1, ..., n$$

Obviously, the next step is to form the Lagrangian:

$$L(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j (x_i^\top x_j) \tag{15}$$

and solve the first order conditions $\frac{\delta L}{\delta b} = 0$, $\frac{\delta L}{\delta w} = 0$ that results in:

$$\sum_{i=1}^{n} \alpha_i y_i = 0 \tag{16}$$

$$w = \sum_{i=1}^{n} \alpha_i y_i x_i = 0 \tag{17}$$

We substitute back in $L(w, \alpha^\star, \alpha)$ and maximize it:

$$L(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j (x_i^\top x_j) \tag{18}$$

subject to:

$$\alpha_i \geq 0$$

$$\sum_{i=1}^{n} \alpha_i y_i = 0$$

It is clear that we have a case of quadratic programming, where the variable to be

optimized is α. The x vector is the input vector and the y vector is the associated label vector.

Finally, the decision will look like this:

$$D(x) = sign\left\{\sum_{j=1}^{n} \alpha_j y_j (x_j^\top x) + b\right\} \tag{19}$$

where x is a new test observation, x_j represents the training data observations that were used to obtain the α-s , y_j are the labels for the training data we used and the α-s are the coefficients that we obtain from the quadratic optimization.

In practice we find that some of the α-s are 0 and others differ from 0. The α-s that are 0 correspond to the non-support vectors and the ones, that are different from 0, correspond to the support vectors.

Because the data may be intermeshed into a low dimensional space, we map it into a higher dimensional space $x_i \rightarrow \Phi(x_i)$, to obtain separable data as shown in Figure 3.

In the optimization function $L(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j (x_i^\top x_j)$, the x-s appear as an inner product, so the mapping will look like this:

$$x_i^\top x_j \rightarrow \Phi(x_i)^\top \Phi(x_j) \tag{20}$$

The higher dimensional space is also called *Feature Space* and must be a *Hilbert Space*, since in this space the concept of inner product applies.

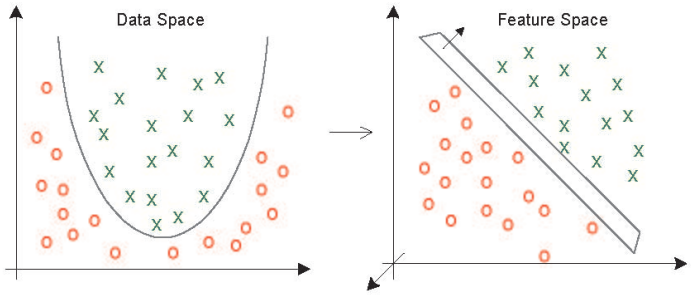


Figure 3: Mapping from a two-dimensional data space into a three-dimensional space of features using a quadratic kernel function $K(x_i, x_j) = (x_i^\top x_j)^2$. The three features correspond to the three components of a quadratic form: $\tilde{x}_1 = x_1^2$, $\tilde{x}_2 = \sqrt{2}x_1x_2$ and $\tilde{x}_3 = x_2^2$, thus, the transformation is $(x_1, x_2) = (x_1^2, \sqrt{2}x_1x_2, x_2^2)$

However, we do not have to know what the mapping function is, since by choosing a kernel we implicitly define the form of the mapped inner product:

$$\Phi(x_i)^\top \Phi(x_j) = K(x_i, x_j) \tag{21}$$

The kernel is therefore the inner product between mapped pairs of points in *Feature Space* and fullfills the *Mercer* conditions of being symmetric and semi positive for mapping low dimensional data into higher dimensional space.

In order to obtain the results we have to take the following steps:

1. Given the binary classified data we choose the kernel function $K(x_i, x_j)$

2. We maximize:

$$L(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j K(x_i, x_j) \tag{22}$$

subject to

$$\alpha_i \geq 0$$

$$\sum_{i=1}^{n} \alpha_i y_i = 0$$

3. Find the bias or trashhold:

$$b = \frac{1}{2} \left\{ min \left[\sum_{i|y_i=+1} \alpha_i y_i K(x_i, x_j) \right] + max \left[\sum_{i|y_i=-1} \alpha_i y_i K(x_i, x_j) \right] \right\} \tag{23}$$

4. Compute the value of the decision function for a new observation

$$D(x) = sign \left[\sum_{i=1}^{n} \alpha_i y_i K(x_i, x) + b \right] \tag{24}$$

Even if we obtained the decision function, data may contain noise, as we have seen in Figure 2. Therefore we have to allow for training errors prior to introducing kernels and we get the following optimisation problem:

$$\min_{w,b,\zeta} \frac{1}{2} w^\top w + C \sum_{i=1}^{n} \zeta_i \tag{25}$$

subject to:

$$y_i(w^\top x_i + b) \geq 1 - \zeta_i$$

$$i = 1, ..., n$$

where ζ is the misclassification error and C is a parameter called capacity that is related to the margin zone. The smaller the C, the greater the margin can be.

We take the same steps:

- form the Lagrangian

$$L(w, b) = \frac{1}{2} w^\top w + C \sum_{i=1}^{n} \zeta_i - \sum_{i=1}^{n} \alpha_i [y_i((w^\top x_i) + b) - 1 + \zeta_i] - \sum_{i=1}^{n} \mu_i \zeta_i \quad (26)$$

  subject to

$$\alpha_i \geq 0$$

$$\mu_i \geq 0$$

- take the derivatives with respect to w, b, ζ, substitute back and obtain the following form which is to be maximized:

$$L(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j K(x_i, x_j) \quad (27)$$

subject to

$$0 \leq \alpha_i \leq C$$

$$\sum_{i=1}^{n} \alpha_i y_i = 0$$

As we can see from the optimization problem, the only difference between not allowing and allowing for training errors is that in the later case the α-s will be constraint by the capacity parameter C.

Moving on, we obtain the following score function:

$$f(x) = \sum_{i=1}^{n} \alpha_i y_i K\left(x, x_i\right) + b \tag{28}$$

We choose K to be a Gaussian kernel

$$K\left(x, x_i\right) = exp\left\{-\left(x - x_i\right)^{\top} r^{-2} \Sigma^{-1} (x - x_i)/2\right\} \tag{29}$$

where r is the coefficient related to the complexity of the classifying functions (the higher the r, the lower is the complexity) and Σ is the variance-covariance matrix of the training data.

3 Dataset Description and Manipulation

3.1 Dataset

The dataset used in our analysis is the Credit reform database and contains balance sheet and income statement information about 20000 solvent and 1000 insolvent German companies. The period spans from 1996 to 2002 and in the case of the insolvent companies the information is gathered 2 years before the insolvency took place. A number of 25 financial indicators were created, denoted as $x_1 \dots x_{25}$. The indicators are presented in Table 1. For the x_9 formula INGA and LB mean intagible assets and lands & buildings, respectively.

A short view of the indicators regarding the quartiles and the median of the solvent and insolvent companies is presented in Table 2. A total number of 25 indicators were used in the analysis.

In order to reduce the effect of the outliers on the results, all observations that exceeded the upper limit of Q75+1.5*IQ (Inter-quartile range) or the lower limit of Q25-1.5*IQ were replaced with these values, see Table 2.

3.2 Validation

In order to perform the analysis, the data is split into 2: training data, containing observations from 1997 to 1999 and validation data containing observations from 2000 to 2002. Using the bootstrap method, 250 default companies and 250 solvent companies are randomly selected and used to obtain the Lagrangian multipliers (the α-s). This operation is perform 10 times. For each set of Lagrangian multipliers,

Ratio	Formula	Description
x1	NI/TA	Net Income/Total Assets
x2	NI/Sales	Net Income/Sales
x3	OI/TA	Operating Income/Total Assets
x4	OI/Sales	Operating Income/Sales
x5	EBIT/TA	EBIT/TA
x6	(EBIT+AD)/TA	(EBIT+AD)/TA
x7	EBIT/Sales	EBIT/Sales
x8	Equity/TA	Equity/Total Assets
x9	(Equity - ITGA)/ (TA-ITGA-Cash-LB)	(Equity- ITGA) / (TA - ITGA - Cash - L&B)
x10	CL/TA	Current Liabilities/Total Assets
x11	(CL-Cash)/TA	(Current Liabilities - Cash)/Total Assets
x12	TL/TA	Total Liabilities/Total Assets
x13	Debt/TA	Debt/Total Assets
x14	EBIT/Interest Expenses	EBIT/Interest Expenses
x15	Cash/TA	Cash/Total Assets
x16	Cash/CL	Cash/Current Liabilities
x17	QA/CL	Quick Assets/Current Liabilities
x18	CA/CL	Current Assets/Current Liabilities
x19	WC/TA	Working Capital/Total Assets
x20	CL/TL	Current Liabilities/Total Liabilities
x21	TA/Sales	Total Assets/Sales
x22	INV/Sales	Inventories/Sales
x23	AR/Sales	Accounts Receivable/Sales
x24	AP/Sales	Accounts payable / Sales
x25	Log(TA)	Log(Total Assets)

Table 1: Variables

Var	All			Insolvent			Solvent		
	Q25	Q50	Q75	Q25	Q50	Q75	Q25	Q50	Q75
x1	0.000	0.014	0.054	-0.030	0.014	0.054	0.000	0.015	0.056
x2	0.000	0.008	0.034	-0.017	0.008	0.034	0.000	0.008	0.035
x3	0.000	0.030	0.090	-0.039	0.030	0.090	0.001	0.032	0.093
x4	0.000	0.017	0.058	-0.024	0.017	0.058	0.001	0.019	0.061
x5	0.004	0.043	0.097	-0.019	0.043	0.097	0.005	0.045	0.100
x6	0.040	0.097	0.171	0.019	0.097	0.171	0.042	0.099	0.174
x7	0.004	0.027	0.069	-0.012	0.027	0.069	0.005	0.028	0.071
x8	0.057	0.175	0.366	0.005	0.175	0.366	0.063	0.183	0.375
x9	0.055	0.183	0.398	0.000	0.183	0.398	0.060	0.193	0.408
x10	0.100	0.302	0.556	0.317	0.302	0.556	0.093	0.292	0.544
x11	0.007	0.231	0.495	0.260	0.231	0.495	0.002	0.220	0.484
x12	0.182	0.500	0.766	0.475	0.500	0.766	0.171	0.486	0.758
x13	0.000	0.122	0.318	0.071	0.122	0.318	0.000	0.117	0.314
x14	0.381	1.885	6.250	-0.814	1.885	6.250	0.443	1.967	6.632
x15	0.004	0.028	0.103	0.002	0.028	0.103	0.004	0.029	0.106
x16	0.011	0.085	0.347	0.004	0.085	0.347	0.012	0.089	0.368
x17	0.616	1.021	1.822	0.438	1.021	1.822	0.630	1.048	1.880
x18	1.074	1.566	2.716	0.974	1.566	2.716	1.082	1.590	2.776
x19	0.057	0.249	0.506	0.000	0.249	0.506	0.060	0.254	0.511
x20	0.555	0.877	1.000	0.620	0.877	1.000	0.550	0.880	1.000
x21	0.351	0.578	1.171	0.401	0.578	1.171	0.348	0.576	1.188
x22	0.015	0.079	0.181	0.062	0.079	0.181	0.014	0.076	0.175
x23	0.043	0.090	0.143	0.064	0.090	0.143	0.042	0.089	0.142
x24	0.032	0.064	0.114	0.084	0.064	0.114	0.031	0.061	0.109
x25	14.280	15.690	17.270	13.903	15.690	17.279	14.312	15.750	17.351

Table 2: Variable description

250 default companies and 250 solvent from the training data are 30 times randomly chosen and used to obtain the SVM and Logit scores.

3.3 Performance Indicators

The scores are used afterwards to obtain the accuracy ratio which is the main indicator used in our SVM vs Logit comparison. In order to obtain the accuracy ratio, first we draw the receiver operating characteristics (ROC) curve as it is shown in Figure 4 and compute the area under the curve (AUC). In case of perfect separation, the ROC curve will look like the blue line, but since the scores of the solvent and insolvent companies are more or less intermeshed, the curve will look like the red curve. In case of a naive model, the roc curve is simply the bisector.

Then we simply apply the formula and get the accuracy:

$$AR = 2\int_1^0 y(x)dx - 1$$

The accuracy ratio will take values between 0, when the ROC curve is the bisector, and 1, when we have a perfect separation case.

3.4 Predictor Selection

Since not all financial indicators are good predictors or not all financial indicators lead to the best accuracy ratio, we need to eliminate the bad indicators or get the

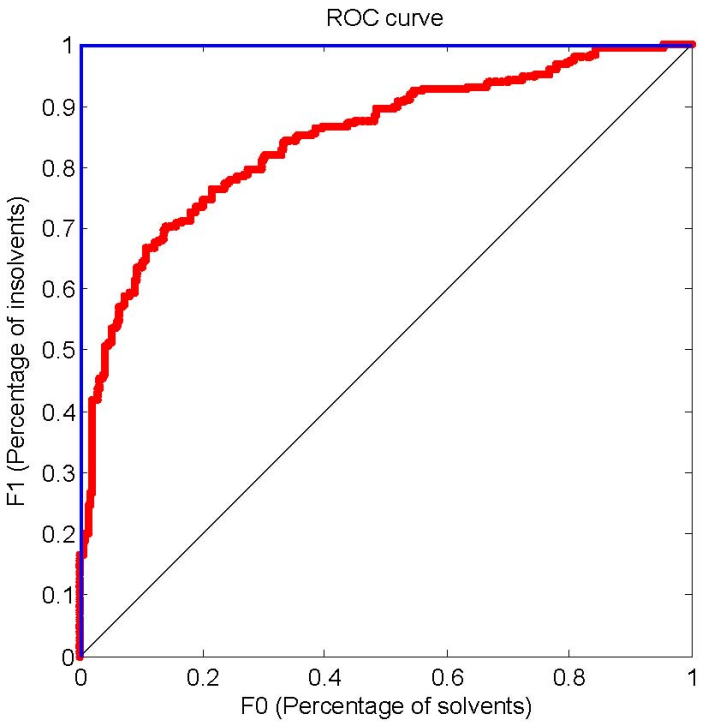


Figure 4: Receiver Operating Characteristics Curve

good indicators. You probably realised that there are two methods of doing that. The first method is called the forward stepwise selection and starts with the indicator with the highest accuracy and then stepwise adds the second best indicator with respect to the accuracy. The second method is called backward elimination and starts with all indicators and eliminates, one by one, the worst indicators in terms of accuracy.

3.5 The Dependency of the Accuracy Ratio (AR) on the C and r Parameters and Other Performance Indicators

After obtaining the best predictor indicators in terms of accuracy we change the C and r to see the dependency of the accuracy on C and r. Besides the accuracy

ratio indicator we draw the probability density functions of the solvent and insolvent companies and compute the first and second type error, which we use as a second best accuracy measure. The first type error refers to the percentage of default companies that are classified as solvent and the second type error refers to the percentage of solvent companies that are classified as insolvent.

3.6 Obtaining Probability of Default (PD) Classes

Obtaining the probabilities of default is the final step of the analysis. Here we take the interval between the default observation with the best (lowest) score and the intersection point of the two probability density functions and divide it into eleven bins. Then we see how many observations fall into each bin and then divide it by the total number of observations. We do this for fifty validation samples. The expectation would be that the probabilities of default obtained in this way would be monotonous, but this is not always the case. Therefore, in order to smooth the results we used a rlowess (locally weighted scatter plot smooth using least squares quadratic polynomial fitting that is resistant to outliers) regressor. There are 2 ways of obtaining the final PD classes using the regressor: one would be to smooth and then average the PD classes and the second method would be to average the PD classes and then smooth.

3.7 Computing the PD for Companies

Having PD classes one may actually compute the score for a new company and see to which PD class the score belongs to. To get the score for a company, the 8 financial indicators for the company were computed using the balance sheets of the

company and implemented in the program. In order to get a representative score the median of 10 bootstrap results was taken.

4 Empirical Results

4.1 SVM vs Logit

Using the backward elimination method, by starting with 25 variables and by considering stepwise all possible combinations, we eliminate one by one the financial indicators whose absence give us the best accuracy ratio and obtain in the end Figure 5:

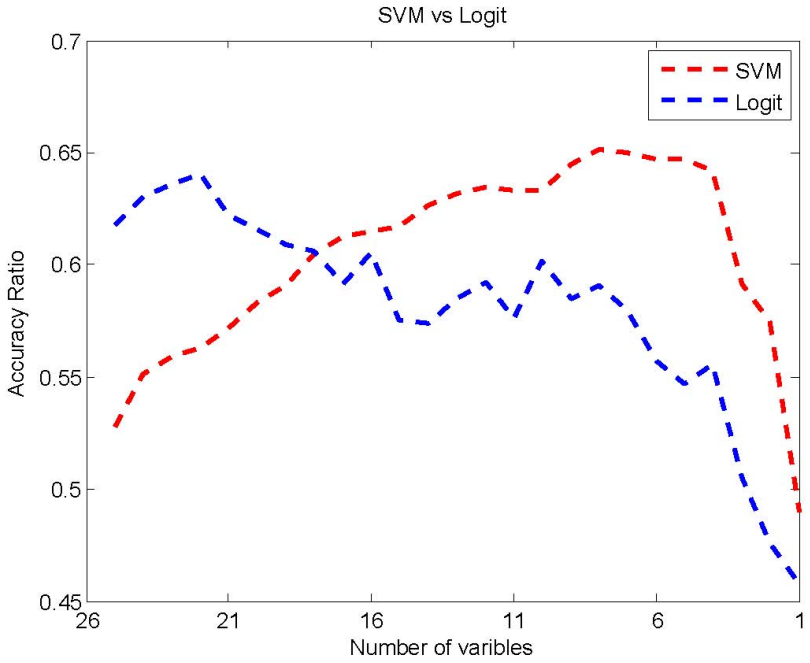


Figure 5: SVM vs Logit

At the beginning the Logit model gives better estimates, but after eliminating some of the variables with low accuracy power the SVM outperforms the Logit. The peak is reached when we have 8 variables: $\frac{NI}{TA}+$ $\frac{OI}{Sales}+$ $\frac{EBIT+DA}{TA}$ $+$ $\frac{CL}{TA}-$ $\frac{Cash}{TA}+$ $\frac{Inv}{Sales}-$ $\frac{AP}{Sales}-$ $\log(TA)_+$. The plus means that the higher the indicator the better the score and

Model	C	r	AR	α	β	Av. Error
SVM	1	1	0.567	0.309	0.242	0.275
	1	2	0.640	0.251	0.266	0.258
	5	1	0.585	0.308	0.224	0.266
	5	2	0.638	0.240	0.279	0.259
	10	1	0.585	0.308	0.224	0.266
	10	2	0.651	0.238	0.273	0.255
	15	1	0.585	0.308	0.224	0.266
	15	2	0.640	0.251	0.266	0.258
	100	1	0.565	0.311	0.234	0.272
	100	2	0.640	0.235	0.284	0.259
Logit			0.584	0.198	0.364	0.281

Table 3: Dependency of AR on C and r

implicitly lower probability of default, whereas the minus represent the opposite: the lower the indicator the better the score and implicitly lower probability of default.

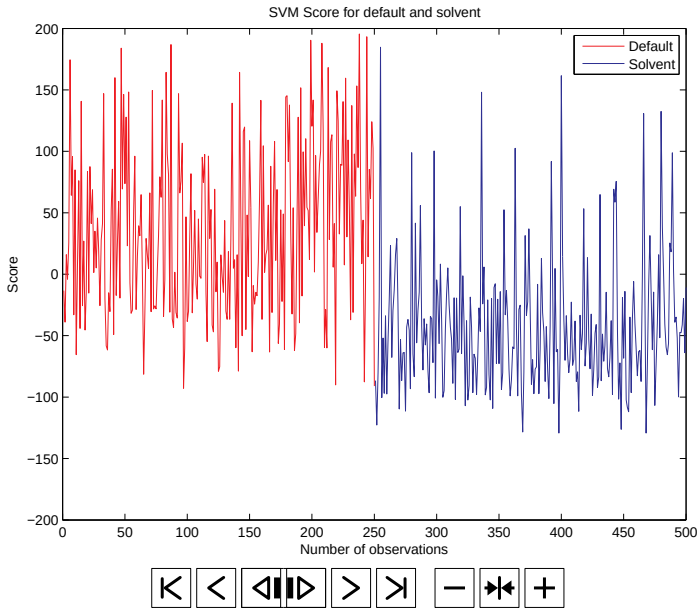
4.2 AR Dependency on C and r

Having obtained the best predictors, we change the C and r to see the dependency as shown in Table 3. Originally, given previous studies, the C and r were taken as 10 and 2 respectively.

As the table shows the best accuracy ratio and the least average error is given by $C = 10$ and $r = 2$.

4.3 SVM Scores

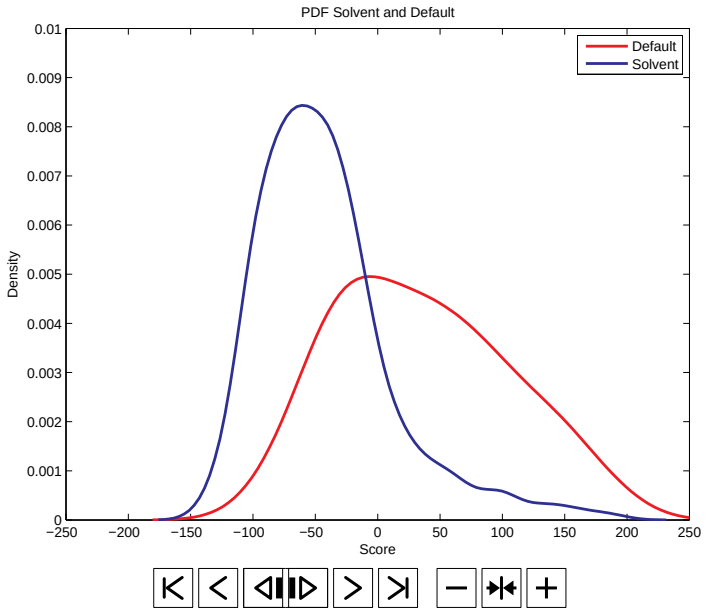
The scores obtained using the best predictors on the validation data are presented in the following movie:



It is obvious that there is a difference between the scores of the default companies and of the solvent ones, namely the scores of the default companies are in most of the cases greater than 0, whereas the scores of the solvent companies are lower than 0. This gives a first picture of the separation property of the SVMs.

4.4 Probability Density Functions of the Scores

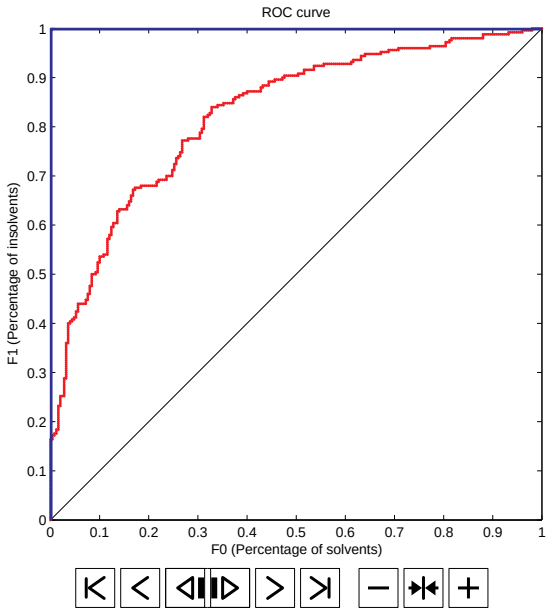
To see better the difference between solvent and insolvent companies, probability density function plots of the scores were created, shown in the following movie:



As expected the pdf-s differ, having their intersection point around zero. The first and second type error are traceable using these graphs, namely, the first type error is the area under the red curve from the minima to the intersection point, while the type two error is the area under the blue curve, from the intersection point to the maxima.

4.5 Receiver Operating Characteristic (ROC) Curves

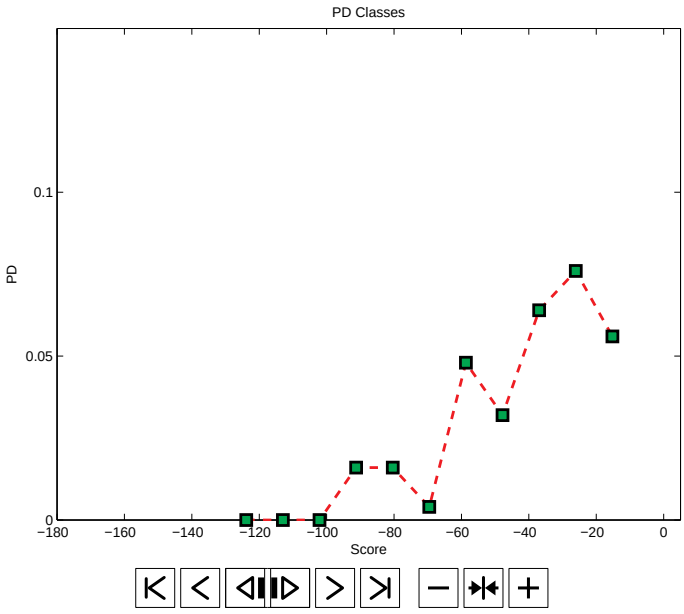
The ROC curves indicate whether the classification holds or not.



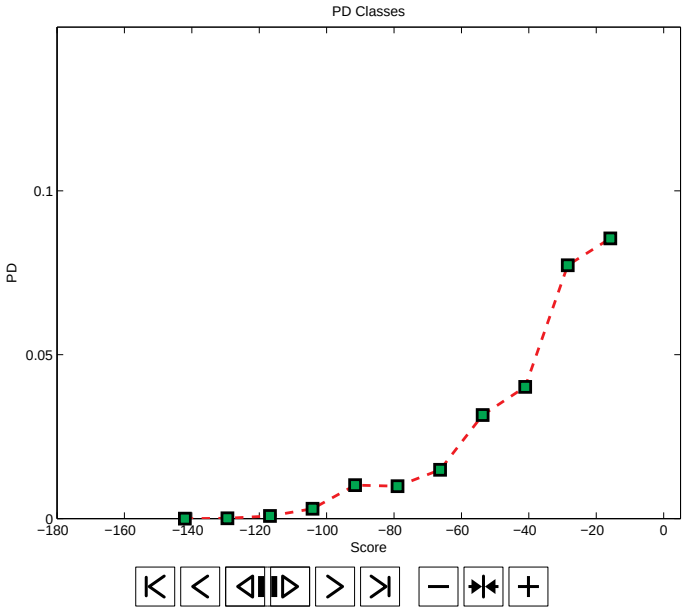
It is visible from the graphs that we obtained a good classification with an area under the ROC curve (AUC) of approximately 0.825.

4.6 Probability of Default Classes

The PD classes for a number of ten validation samples are presented in the movie below. The first method was used, namely to smooth and then average the classes. The final graph of the movie shows the final PD classes.



In the next movie the second method is used: to average first the classes and then smooth them. Therefore we have only four graphs, the fourt graph being the one that has the final PD classes.



To see the difference between the two methods a plot containing the final two PD classes was constructed.

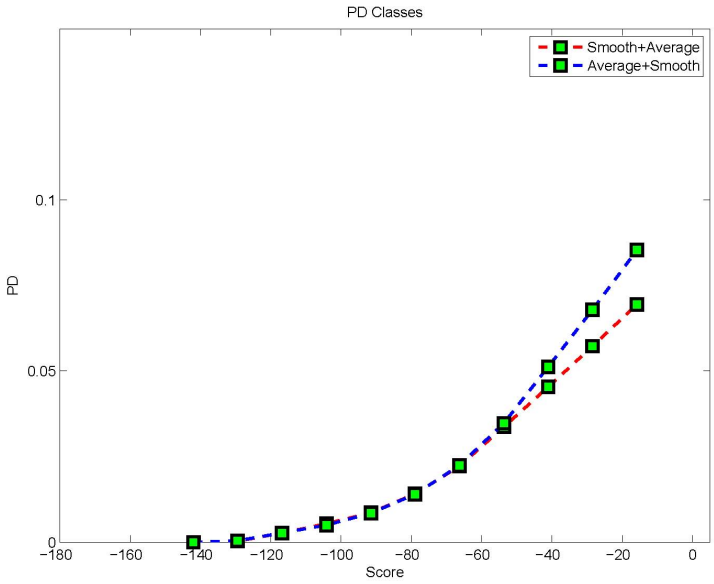


Figure 6: SVM vs Logit

The second method of averaging and than smoothing gives higher probabilities of default.

4.7 PDs for Companies

After completing all the steps we can finally see what the PDs are for other companies, such as DAX companies. It is exciting to see what the results look like.

Four random companies were taken: MAN, Henkel, BASF and Altana for 3 different years: 2000, 2001 and 2008. The first 2 year match the period of the validation data and the third one was taken to see whether there were differences in time between the results.

Firstly, the scores that we got were negative which means they belong to the solvent group. Next, we see to which PD class they belong and write the result accordingly.

Company	Year	Score	PD1	PD2
MAN	2000	-10	0.0695	0.0853
MAN	2001	-11	0.0695	0.0853
MAN	2008	-23	0.0572	0.0679
Henkel	2000	-25	0.0572	0.0679
Henkel	2001	-28	0.0572	0.0679
Henkel	2008	-20	0.0695	0.0853
BASF	2000	-26	0.0572	0.0679
BASF	2001	-29	0.0572	0.0679
BASF	2008	-34	0.0454	0.0512
Altana	2000	-29	0.0572	0.0679
Altana	2001	-28	0.0572	0.0679
Altana	2008	-26	0.0572	0.0679

Table 4: PDs for different companies

Since the information regarding the default companies is taken 2 years prior to the default of the company, we may say that in our case the final results represent the two year default probability.

5 Conclusions

A non-parametric technique called "Support Vector Machines" was used in order to analyse solvent and insolvent German companies and to come up with a separation decision function that would accurately classify new data. We used the bootstrap method to select the observations and the backward elimination technique to get rid of the variables with low accuracy power. The SVM was compared to the Logit model and the results indicate that the SVM performs betterthan the Logit. In the comparison the accuracy ratio was used as main selection indicator. We started with 25 predictor variables and found eight best predictors: $\frac{NI}{TA}$, $\frac{OI}{Sales}$, $\frac{EBIT+DA}{TA}$, $\frac{CL}{TA}$, $\frac{Cash}{TA}$, $\frac{Inv}{Sales}$, $\frac{AP}{Sales}$, $\log(TA)$. After finding the best predictor variables we added the misclassification error indicator to select the best C and r parameters that influence our results. After finding the optimal parameters, we analysed the results of the validation data. The sample scores show that default companies have normally scores higher than zero, while solvent companies have scores lower than zero. This was underlined by plotting the probability density functions of the scores. Moreover, as another measure of separation that is directly related to the accuracy ratio, the receiver operating characteristics curve was computed that shows good separation. Finally, we divided the interval to the right of the intersection point of the probability density functions into 11 equal bins and computed the percent of the default companies that fall into each bin. This shows what the probability of default of a company with a score corresponding to that bin is. Each bin was considered a class of default probability. Since classes were not monotonous, we used a locally weighted least squares quadratic polynomial fitting regressor resistant to outliers to smooth the data. Finally, we computed scores for 4 randomly chosen DAX companies and assigned the corresponding PDs to each one.

References

Banasik J., Crook J. N., Thomas L. C. (1999), Not if but When will Borrowers Default, *The Journal of the Operational Research Society*, UK

Bank for International Settlements (2005), An Explanatory Note on the Basel II IRB Risk Weight Functions, *Press & Communications*, Basel

Belloti T., Crook J. (2008), Support vector machines for credit scoring and discovery of significant features, Science Direct

Bluhm C., Overbeck L., Wagner C. (2002), *An Introduction to Credit Risk Modeling*, Chapman & Hall/CRC, London

Chen S., Härdle W., Moro R. (2006), Estimation of Default Probability with Support Vector Machines, *SFB 649 Discussion Paper*, Berlin

Cizek AP., Härdle W. (2005), *R. Weron, Statistical Tools for Finance and Insurance*, Springer, Berlin

Friedman C. (2002), CreditModel Technical White Paper, *Standard&Poor's*, New York

Gestel T. Van , Baesens B., Garcia I. J., Dijcke P. Van, (2003), A Support Vector Machine Approach to Credit Scoring, *DefaultRisk.com Papers*

Härdle W. , Moro R., Schäfer D. (2005), Bancruptcy Analysis with Support Vector Machines, *SFB 649 Discussion Paper*, Berlin

Härdle W. , Moro R., Schäfer D. (2007), Estimating probabilities of default with

support vector machines, *Deutsche Bundesbank Discussion Papers*, Berlin

Moro R.A. , Härdle W., Schäfer D. (2006), Estimating Probabilities of Default with Support Vector Machines, M.Sc. Master Thesis in Statistics, Humboldt University

Vapnik V. (1995), *The Nature of Statistical Learning Theory*, Springer, Berlin

Vaclavik T. (2007), Probability of Default Basic Methods Overview, Institute for Economics Studies, UK

http://videolectures.net/epsrcws08_campbell_isvm/

http://videolectures.net/mlss06tw_lin_svm/

List of Figures

List of Tables

Declaration of Authorship

I hereby certify that the thesis I am submitting is entirely my own original work except where otherwise indicated. I am aware of the University's regulations concerning plagiarism, including those regulations concerning disciplinary actions that may result from plagiarism. Any use of the works of any other author, in any form, is properly acknowledged at their point of use.

Sebe-Vodislav Razvan-Alexandru

Berlin, 13. August, 2009